Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) A maximum likelihood decoder comprising:

a first probability computing means for computing a logarithm of branch metric (γ), which is a logarithm of probability of a particular branch of a Trellis diagram, computed only based on the knowledge of input and output symbols associated with the particular branch;

a second probability computing means for computing a logarithm of forward state metric (α), which is a logarithm of probability of a particular state of the Trellis diagram, given the probabilities of states at previous time instances;

a third probability computing means for computing a logarithm of backward state metric (β), which is a logarithm of probability of the particular state of the Trellis diagram, given the probabilities of states at future time instances,

wherein each of said second probability computing means and said third probability computing means includes

a path selection means, said path selection means including:

a plurality of comparator circuits,

a plurality of absolute value computation circuits, and

a plurality of selectors to select ~~for obtaining~~ at least two paths ~~of getting~~ to a ~~decoding~~ state in the Trellis diagram from at least three paths~~, and for selecting a maximum likelihood path from said at least two paths, wherein a log likelihood of getting to a state in the decoder is determined by an input value from an encoder encoded with a trellis so as to provide at least three paths for getting to the decoding state, said input value received through a communication~~

~~channel having noise such that said input value is regarded as being a soft value~~ said plurality of selectors configured to enable concurrent operations of addition, comparison, and selection in log likelihood computations; and

a soft-output determining means for determining a log soft-output logarithmically expressing a soft-output in each time slot, given said forward and backward state metrics as well as said branch metric.

2. (Currently Amended) The decoder according to claim 1, wherein each of said ~~path selection means includes a comparison means~~ plurality of comparator circuits includes

a ~~for comparing~~ log likelihood~~s~~ comparator for comparing log likelihood of all combinations of said at least two paths selected from said at least three paths ~~for getting to each state~~.

3. (Currently Amended) The decoder according to claim 2, ~~further comprising:~~ wherein each of said ~~an~~ plurality of absolute value ~~selection means~~ computation circuits includes

a computation means for ~~selecting an absolute value of the difference between data corresponding to the maximum likelihood path and data corresponding a second maximum likelihood path~~ computing absolute values of the difference of said at least two paths selected from said at least three paths.

4. (Currently Amended) The decoder according to claim 3, wherein each of said plurality of selectors includes

~~absolute value selection means includes an absolute value computing means for computing absolute values of the difference of all combinations of said at least two paths selected from said at least three paths,~~

~~wherein the computed absolute values are compared for magnitude according to outcome of said comparison means~~

an absolute value selection means for selecting an absolute value of the difference between data corresponding to a maximum likelihood path and data corresponding to a second maximum likelihood path selected from said at least three paths.

5. (Currently Amended) The decoder according to claim ~~3~~ 4, further comprising:

a linear approximation means for computing by linear approximation a correction term added to obtain said log likelihood and expressed by a one-dimensional function relative to a variable, said linear approximation means using said variable as an absolute value of the difference between data corresponding to said maximum likelihood path and fed from said absolute value selection means and data corresponding to said second maximum likelihood path.

6. (Previously Presented) The decoder according to claim 5, wherein said linear approximation means computes said correction term by expressing a coefficient representing the gradient of said one-dimensional function for multiplying said variable at least by means of a power exponent of 2.

7. (Original) The decoder according to claim 6, wherein said linear approximation means discards lower bits of an input data according to the power exponent expressing the coefficient representing the gradient of said function.

8. (Previously Presented) The decoder according to claim 6, wherein said linear approximation means discards bits from the lowest bit to the k-th lowest bit of the input data when the coefficient representing the gradient of said function is expressed by -2^{-k}.

9. (Previously Presented) The decoder according to claim 6, wherein said linear approximation means computes said correction term by expressing the coefficient representing an intercept of said function by means of a power exponent of 2.

10. (Original) The decoder according to claim 9, wherein said linear approximation means computes said correction term by expressing the coefficient representing the intercept of said function by means $2^{m} - 1$.

11. (Previously Presented) The decoder according to claim 10, wherein said linear approximation means discards bits from the lowest bit to the k-th lowest bit of the n-bit input data and inverts m bits from the k+1-th lowest bit to the m+k-th lowest bit when the coefficient representing the gradient of said function is expressed by means of -2^{-k}.

12. (Original) The decoder according to claim 6, wherein said correction term shows a positive value.

13. (Original) The decoder according to claim 12, wherein said linear approximation means makes the correction term equal to 0 when a negative value is produced by computing said correction term.

14-48. (Canceled)

49. (New) A maximum likelihood decoding method comprising:

first computing a logarithm of branch metric (γ), which is a logarithm of probability of a particular branch of a Trellis diagram, computed only based on the knowledge of input and output symbols associated with the particular branch;

second computing a logarithm of forward state metric (α), which is a logarithm of probability of a particular state of the Trellis diagram, given the probabilities of states at previous time instances;

third computing a logarithm of backward state metric (β), which is a logarithm of probability of the particular state of the Trellis diagram, given the probabilities of states at future time instances,

wherein each of said second computing and said third computing includes comparing, absolute value computing, and selecting to enable concurrent operations of addition, comparison, and selection in log likelihood computations through selection of at least two paths to a state in the Trellis diagram from at least three paths; and

determining a log soft-output logarithmically expressing a soft-output in each time slot, given said forward and backward state metrics as well as said branch metric.

50. (New) The method according to claim 49, wherein said comparing includes

comparing log likelihood of all combinations of said at least two paths selected from said at least three paths.

51. (New) The method according to claim 50, wherein said absolute value computing includes

computing absolute values of the difference of all combinations of said at least two paths selected from said at least three paths.

52. (New) The method according to claim 51, wherein said selecting includes

selecting an absolute value of the difference between data corresponding to a maximum likelihood path and data corresponding to a second maximum likelihood path from said at least three paths.

53. (New) The method according to claim 52, further comprising:

computing by linear approximation a correction term added to obtain said log likelihood and expressed by a one-dimensional function relative to a variable; and

using said variable as an absolute value of the difference between data corresponding to said maximum likelihood path and fed from said absolute value selection means and data corresponding to said second maximum likelihood path.

54. (New) The method according to claim 53, wherein said computing by linear approximation computes

said correction term by expressing a coefficient representing the gradient of said one-dimensional function for multiplying said variable at least by means of a power exponent of 2.

55. (New) The method according to claim 54, wherein said computing by linear approximation discards

lower bits of an input data according to the power exponent expressing the coefficient representing the gradient of said function.

56. (New) The method according to claim 54, wherein said computing by linear approximation discards

bits from the lowest bit to the k-th lowest bit of the input data when the coefficient representing the gradient of said function is expressed by -2^{-k}.

57. (New) The method according to claim 54, wherein said computing by linear approximation computes

said correction term by expressing the coefficient representing an intercept of said function by means of a power exponent of 2.

58. (New) The method according to claim 57, wherein said computing by linear approximation computes

said correction term by expressing the coefficient representing the intercept of said function by means $2^m - 1$.

59. (New) The method according to claim 58, wherein said computing by linear approximation discards

bits from the lowest bit to the k-th lowest bit of the n-bit input data and inverts m bits from the k+1-th lowest bit to the m+k-th lowest bit when the coefficient representing the gradient of said function is expressed by means of -2^{-k}.

60. (New) The method according to claim 54, wherein said correction term shows a positive value.

61. (New) The method according to claim 60, wherein said computing by linear approximation makes

the correction term equal to 0 when a negative value is produced by computing said correction term.